Towards a Meta-Analysis-Based User Assistant for Analysis Processes

Julien Aligon, William Raynaut, Philippe Roussille,
Chantal Soulé-Dupuy, Nathalie Vallès-Parlangeau

IRIT Lab, University of Toulouse
Toulouse, France

e-mail: firstname.surname@irit.fr

ABSTRACT

We propose a user assistant to devise analysis processes, based on a meta-analysis approach. Especially, we described a recommender system in three steps and relying on past analysis. We tested the performance of our approach with other classical methods found in the literature. The result shows that our proposal outperforms the other ones.

Keywords

Data Analysis, Data-Mining, Analysis Workflow, Meta-Analysis

1. INTRODUCTION

The emergence of the big data phenomenon has led to increasing demands in data analysis, which most often are conducted by experts of their respective application fields (not necessarily in data science). An assistance to those users is essential for designing and applying analysis processes to their problems. Several assistants for data analysis were proposed over the years in order to allow such end-users to perform useful data analysis. Their primary function is to bring some order of automation into the *Meta-analysis* process.

Meta-analysis designates the very general task of finding an efficient (or most efficient) way to solve a given data analysis problem. As such, it covers a very wide range of tasks, a good many of which have already been extensively studied. For instance, if we consider the very specific problem of Boolean Satisfiability (SAT), we can find different approaches, such as [17], based on the selection of a most efficient algorithm to solve a particular problem instance. Such approaches are designated as *portfolio* for the SAT problem, but have equivalents on many other problems. Their most common denomination would be *algorithm selection* methods, many of which have been studied for machine learning problems, such as classification [7], regression [5], or instance selection [8]. These many different problems have been well studied on their own, but the next step for *meta-analysis* research is to start unifying some of them. In particular, the problem of *data analysis workflow recommendation* has received an increased interest over the last few years [9, 13, 14, 18]. Most of the papers working on the recommendation field (including in a variety of contexts) assume that the best results are generally given by collaborative filtering systems [1, 4]. It consists in the elicitation of past workflows (sequences of operators allowing to mine the data) solving a range of different data analysis problems, but remains mostly focused on predictive modelling.

Using meta-analysis principle, we focus this paper on a first approach of user assistant for analysis processes. The novelty of our approach is to propose a user assistant taking into account all features characterizing the processes of past workflows:

- The dataset which had to be analyzed
- The type of indicators expected over the dataset (examples: accuracy, recall, etc.)
- The sequence of operators leading to mine the dataset

The difficulty of our approach is to exploit and connect these three features in a unique user assistant. Thus, considering a current dataset and expected results on this one, the user assistant will propose one of the most relevant past workflow helping a current user to improve his/her analysis.

The paper is organized as follows. Section 2 defines our meta-analysis-based framework for recommending workflows. Section 3 presents experimental results of our approach based on a real use-case. Section 4 concludes our work and details several perspectives.

2. META-ANALYSIS FOR WORKFLOW RECOMMENDATION

In this section, we describe our workflow recommender system using a meta-analysis-based approach. This approach is depicted in Figure 1.

2.1 Toy example

Let us illustrate the elements of Figure 1 over a simple example.

Dr. Flour is a flower biologist doing research over the different species of plants she has; namely iris setosa, iris virginica and iris versicolor. She's trying to come up with an efficient model of classification given, among other criterion, the petal and the sepal size. Now, Dr. Flour has no real expertise in the field of data mining, so she does not know really what possible steps are available to her. At the same time, Dr. Kaktus is a famous geneticist who, with a few of his colleagues, came up with a novel computer model which can classify cacti flowers based on their size.

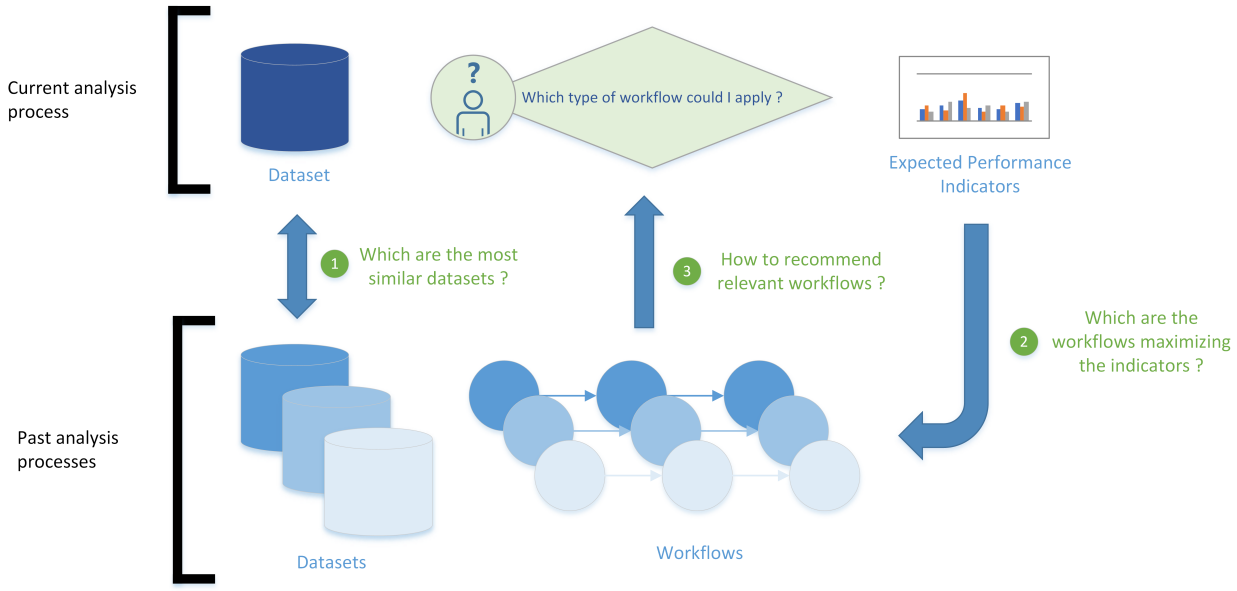

Figure 1. Global steps of the meta-analysis process

Fortunately, Dr. Kaktus uploaded his data and his analysis through the platform for anyone to evaluate or run operations on, alongside the various versions of his model (i.e., several workflows). So, our system will be able to select the features and the desired preferences on the model, and get a solution which could be tailor-made for Dr. Flour needs.

Dr Flour primarily wants her model to be the most correct, so she is given predictive accuracy as a key criterion with a full weight of 1. She would rather get non-trivial information out of it, and so is advised to use the criterion of Kononenko's *Information Score* [6], measuring the amount of *new* information produced by the model, to be somewhat important. She assigns it a medium weight of 0.5. Ultimately, she remarks that Cohen's *Kappa* [2] could also be useful with its ability to account for the chance of random good guesses from the model, and adds it with a lower weight of 0.1.

2.2 Recommendation process

After illustrating our approach with a toy example, we describe the general process of our recommender system.

Consider the following definition of a workflow and a result of a workflow : Over a dataset D, a workflow w is defined as a sequence of data-mining operators producing a unique result r. r is composed of the data mined on D as well as indicators allowing to estimate the relevance of r.

Considering a current analysis, the systems recommends workflows from past analysis. These workflows are suggested thanks to their relevance with performance indicators expected by the current user on its dataset. More precisely, our meta-analysis-based approach is composed of three steps. Firstly, the system identifies past datasets that are the most similar with the current dataset. Then, from these past datasets, their related workflows are selected and executed on the current dataset. Finally, workflows maximizing the indicators are recommended to the current user.

3. EXPERIMENTS

3.1 Proof of concept

In order to test our results, we build a first proof of concept based on the online repository for data science called OpenML [15]. OpenML is an environment for crowdsourcing data sets and flows. It can be used to extract models or predictions. We chose the Weka [3] environment to work with, as it provides a clear and efficient interface to build workflows in a graphical and user-friendly way.

We worked with Weka Knowledge Flow [3] for editing and running flows with any data, through a user-friendly interface. We restricted ourselves with tasks of supervised classification for the example.

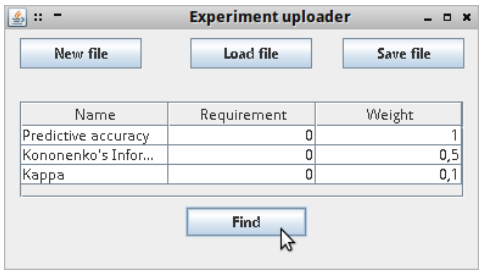

Figure 2. The interface used to set the parameters

Through the aforementioned scenario, Dr Kaktus picks and uploads his flows and his models using the dedicated interface (c.f. Figure 2). Similarly, Dr Flour selects her dataset and her constraints using the experiment window. Once her input is done, the tool runs through the steps of the aforementioned recommendation process (see Section 2.2). The selected workflow is then edited to include Flour's data through Kaktus' model.

3.2 Validation

In order to validate this process, we performed a simple classifier recommendation experiment. We constructed a base of past analysis processes by running 10 well-

Table 1. Performance of the diverse selection techniques

Test dataset	**Domain knowledge**	**Global best**	**Subsampling**	**Meta-analysis**
Electricity	0.460	0.652	0.657	0.998
Spambase	0.966	0.880	0.861	0.880
Iris	0.959	0.969	1.000	0.969
Anneal	0.803	0.852	1.000	0.852
Car	0.876	0.810	0.835	0.810
Mean	**0.813**	**0.833**	**0.871**	**0.902**

known classification algorithms from the Weka API [3] on 10 classification tasks from OpenML. The tasks were picked to consider various classification problems (from binomial to multinomial classification) over a range of very different datasets. The classification algorithms were also picked to exhibit various biases, and cover a range of well-known techniques from the literature. Detailed information about every classifier, dataset, task, criterion and strategy used in this paper are available online[1].

The point is then to use this base of past experiments in order to choose, for 5 new classification problems from OpenML, which of our 10 classification algorithms to use. We will compare the performance in doing this selection of four different methods. These methods, described below, mimic the strategies often used by analysts of different expertise to perform classifier selection (as described in [10] or [11] for instance):

1. **Domain knowledge** : Directly choose a classifier that seems appropriate to domain experts, using some general classification guidelines. For instance, the Naïve Bayes classifier is known to perform well on datasets with good independence among attributes.

2. **Global best** : Check which classifier performed best on the base of past experiments and always use that one.

3. **Subsampling** : Try each classifier on a subsample (100 instances) of the new dataset. This allows to estimate their performance and make an informed decision in choosing the best, while not taking the potentially considerable time required to try each classifier on the full dataset. This method actually "cheats" by experimenting with the actual new dataset, but as it only performs a really simple experiment, it may still be compared with methods using only prior knowledge.

4. **Meta-analysis** : Use the meta-analysis process described in Section 2 to choose the classifier to use.

We evaluate the meta-level performance of each of those methods according to the Meta-Level Evaluation Framework proposed in [12]. We only remind here the meta-level performance criterion from this framework :

[1]See `https://github.com/WilliamRaynaut/CSIT_meta_analysis_assistant`

Definition 1. *Let* $\boldsymbol{x}$ *be the actual value of the objective criterion (accuracy) achieved on* $\boldsymbol{dataset}_i$ *by the classifier* $\boldsymbol{classifier}_j$ *predicted by the recommendation experiment. Let* **best** *be the best value of the objective criterion achieved on* $\boldsymbol{dataset}_i$ *among the classifiers* $\boldsymbol{classifier}_{1..m}$. *Let* **def** *be the actual value of the objective criterion achieved on* $\boldsymbol{dataset}_i$ *by the default classifier (majority class classifier).*
We define the performance of the recommendation of $\boldsymbol{classifier}_j$ *for* $\boldsymbol{dataset}_i$ *:*

$$\mathrm{P}(\mathrm{classifier}_j, \mathrm{dataset}_i) = Max\Big(-1, 1 - \frac{|best - x|}{|best - def|}\Big)$$

As illustrated in Figure 3, this performance criterion reaches its maximum of 1 when the recommended classifier achieves the best accuracy among the studied classifiers, and hits 0 when the predicted classifier achieves the same accuracy as the default classifier. It is bound downward in -1 to avoid distinguishing between the already useless solutions.

Figure 3. Performance of a recommendation

Then, if a recommendation experiment has a performance of 0.9, it means that the recommended classifier is 90% as good as the best available. This allows to directly and easily compare the performance of meta-level recommendation methods, by simply averaging over a range of meta-level experiments. We then experiment our 4 selection strategies on 5 new classification tasks from OpenML, and present their respective performances in Table 1.

We can notice that the **Domain knowledge** strategy leads to inconsistent performances, which is quite natural since it is by nature incomplete and hard to interpret for non-experts. This is one of the major difficulties that domain experts face when attempting data analysis : the knowledge of the meta-domain (the *know-how* to build, test and compare analysis processes) is mostly implicit, and by nature incomplete. Moreover, any such knowledge gathered on the analysis of a particular topic is by no means guaranteed to apply to the analysis of

another, making general guidelines marginally useful at best.

The **Global best** strategy shows a bit better performances by exploiting repeatedly the algorithm that seems to be the most competitive. The main flaw of this strategy is its lack of adaptability : as per Wolpert's *No free lunch* theorems [16], any good performance of a classifier on a given dataset has to be offset by poor performance on another. There literally cannot be a *"best"* classifier in general, but only on restricted subspaces.

The **Subsampling** strategy then leads to acceptable performances, but, as stated before, one must keep in mind that it makes use of *posterior* knowledge, by actually experimenting with the new dataset to make an informed decision, while all other methods use only *prior* knowledge.

Finally, we can see that while the **Meta-analysis**-based recommendation does not necessarily lead to the best performance on each individual test dataset, its performance is on average higher than that of the other methods, reaching the 90% threshold. Its results also appear more consistent than that of the other strategies, suggesting more reliability. Finally, one must keep in mind that while the other strategies make little to no use of the base of prior experiments, the **Meta-analysis** relies very heavily on it, meaning that as this base grows, its performance is bound to improve, while the other strategies offer no such guarantee.

4. CONCLUSION & PERSPECTIVES

In this paper, we presented the first proposal of a user assistant to devise analysis processes. In particular, this assistant is based on a meta-analysis approach. Especially, we described a recommender system in three steps relying on past analysis (using similarities between datasets and expected indicators). We tested the performance of our approach with other classical methods found in the literature. The result shows that our proposal outperforms the other ones.

In order to be the most exhaustive as possible, our main short-term perspectives will consider more data-mining tools such as KNIME, RapidMiner, R or Orange. We will also implement other tasks than classification like regression or clustering. Our long-term perspective will have to adapt, as much as possible, the recommendations to a current user. Indeed, it is essential that this user can easily appropriate the analysis results.

References

[1] Julien Aligon et al. "A collaborative filtering approach for recommending {OLAP} sessions". In: *Decision Support Systems* 69 (2015), pp. 20–30. ISSN: 0167-9236.

[2] Jacob Cohen. "Weighted kappa: Nominal scale agreement provision for scaled disagreement or partial credit." In: *Psychological bulletin* 70.4 (1968), p. 213.

[3] Mark Hall et al. "The WEKA data mining software: an update". In: *ACM SIGKDD explorations newsletter* 11.1 (2009), pp. 10–18.

[4] Jonathan L. Herlocker et al. "Evaluating Collaborative Filtering Recommender Systems". In: *ACM Trans. Inf. Syst.* 22.1 (Jan. 2004), pp. 5–53.

[5] Alexandros Kalousis and Maelanie Hilario. "Model selection via meta-learning: a comparative study". In: *International Journal on Artificial Intelligence Tools* 10.04 (2001), pp. 525–554.

[6] Igor Kononenko and Ivan Bratko. "Information-Based Evaluation Criterion for Classifier's Performance". In: *Machine Learning* 6.1 (Jan. 1991), pp. 67–80. ISSN: 0885-6125.

[7] Rui Leite, Pavel Brazdil, and Joaquin Vanschoren. "Selecting classification algorithms with active testing". In: *Machine Learning and Data Mining in Pattern Recognition*. Springer, 2012, pp. 117–131.

[8] Enrique Leyva, Adriana Gonzalez, and Roxana Perez. "A Set of Complexity Measures Designed for Applying Meta-Learning to Instance Selection". In: *Knowledge and Data Engineering, IEEE Transactions on* 27.2 (2015), pp. 354–367.

[9] Phong Nguyen, Melanie Hilario, and Alexandros Kalousis. "Using meta-mining to support data mining workflow planning and optimization". In: *Journal of Artificial Intelligence Research* (2014), pp. 605–644.

[10] Mary K Obenshain. "Application of data mining techniques to healthcare data". In: *Infection Control* 25.08 (2004), pp. 690–695.

[11] Sellappan Palaniappan and Rafiah Awang. "Intelligent Heart Disease Prediction System Using Data Mining Techniques". In: *Proceedings of the 2008 IEEE/ACS International Conference on Computer Systems and Applications*. AICCSA. IEEE Computer Society, 2008, pp. 108–115.

[12] William Raynaut, Chantal Soule-Dupuy, and Nathalie Valles-Parlangeau. "Meta-Mining Evaluation Framework : A large scale proof of concept on Meta-Learning". In: *29th Australasian Joint Conference on Artificial Intelligence*. Springer, Dec. 5, 2016, pp. 215–228. URL: `https://link.springer.com/book/10.1007/978-3-319-50127-7`.

[13] Floarea Serban et al. "A survey of intelligent assistants for data analysis". In: *ACM Computing Surveys (CSUR)* 45.3 (2013), p. 31.

[14] Quan Sun, Bernhard Pfahringer, and Michael Mayo. "Full model selection in the space of data mining operators". In: *Proceedings of the 14th annual conference companion on Genetic and evolutionary computation*. ACM. 2012, pp. 1503–1504.

[15] Joaquin Vanschoren et al. "OpenML: Networked Science in Machine Learning". In: *SIGKDD Explorations* 15.2 (2013), pp. 49–60. DOI: `10.1145/2641190.2641198`. URL: `http://doi.acm.org/10.1145/2641190.2641198` (visited on 04/01/2017).

[16] David H Wolpert. "The lack of a priori distinctions between learning algorithms". In: *Neural computation* 8.7 (1996), pp. 1341–1390.

[17] Lin Xu et al. "SATzilla2012: improved algorithm selection based on cost-sensitive classification models". In: *SAT Challenge 2012: Solver and Benchmark Descriptions* (2012), pp. 57–58.

[18] Monika Zakova et al. "Automating knowledge discovery workflow composition through ontology-based planning". In: *Automation Science and Engineering, IEEE Transactions on* 8.2 (2011), pp. 253–264.